

А.ВЕКШИН, А.СЕМЕНЮК, А.ДОЛГАРЕВ, НТУ “ХПИ”

О РАЗРАБОТКЕ ПРОТОТИПА ИМИТАЦИОННОЙ СРЕДЫ ДЛЯ СБОРА ТРЕБОВАНИЙ КАЧЕСТВА

В статті розглянута задача реалізації прототипу інтерактивного імітаційного середовища, за допомогою якого користувачі інформаційних систем, що розробляються, будуть мати можливість випробувати різноманітні варіанти взаємодії з імітаційними моделями цих систем з метою надання оцінки різним аспектам якості моделі, які потім будуть використані для розробки вимог якості до системи, що розроблюється. Наведена архітектура інтерактивного середовища, а також можливі шляхи його програмної реалізації.

In this paper, the problem of constructing a prototype of parameterized online simulation environment (POSE). Using this environment, stakeholders can try different approaches to interaction with the simulation models of the system under development (SUD) and make assessments of various aspects of quality of this system. These assessments are supposed to be used for eliciting the quality requirements to the SUD. Some issues in implementing the architecture for this prototype are considered and possible program solutions are discussed.

Введение. Выделение требований качества при разработке программного обеспечения является сложной задачей. Проблема заключается в том, что заинтересованные лица не всегда готовы к тому, чтобы сформулировать требования к качеству предполагаемой системы, если они не сталкивались на практике с системами такого рода. Традиционный подход к разработке требований, который заключается в тесном общении с заинтересованными сторонами, не обязательно является лучшим, а тем более единственным из возможных вариантов. Для разрешения задачи необходимым является поиск такого решения, которое бы отличалось от традиционных технологий сбора требований [1,2].

В качестве альтернативного варианта предлагается идея виртуальной среды, в которую может быть занесена начальная версия функциональных требований к компьютеризированной системе. Эта среда, опираясь на интегрированный имитационный компонент, даст возможность заинтересованным сторонам оказываться в предполагаемых рабочих ситуациях и на основании этого опыта оценивать качество разрабатываемой системы.

В связи с этим предполагается разработать интерактивную имитационную среду [3,4], при помощи которой возможно проводить эксперименты с моделями реальных систем для выявления требований качества, предъявляемых к этим системам. В данной статье рассмотрены вопросы реализации прототипа такой системы.

1. Базовая идея системы. Интерактивная имитационная среда может использоваться в трех различных режимах:

- в режиме адаптации, в котором происходит адаптация среды имитационного моделирования для конкретной задачи;
- в режиме эксперимента, предназначенном для интерактивного сбора требований в ходе взаимодействия заинтересованных лиц с имитационными моделями, для которых определены различные значения ключевых параметров;
- в режиме анализа, который используется для анализа результатов экспериментов и формулирования требований, собранных в ходе этих экспериментов.

Организация рассматривается как система, в которой выполняется определенный набор бизнес-процессов. Как следствие, для создания имитационной модели организации необходимо смоделировать ее структуру, ресурсы (в том числе людские), которые она использует, ее бизнес-процессы, процессы поддержки и процессы управления.

Для моделирования процессов предлагается использовать ассемблер процессов, который сохраняет модели процессов, описанные на различных языках моделирования процессов, и дает возможность отображать модели, представленные с использованием одного языка, на другой язык при условии, что выразительные средства языка позволяют выполнять такое отображение, и в системе зарегистрированы соответствующие драйверы для отображения. Основная цель разработки интерактивной имитационной среды в том, чтобы дать возможность потенциальным пользователям разрабатываемой системы получить опыт взаимодействия с имитационной моделью, которая опирается на ключевые параметры использования системы.

Одной из наиболее важных задач, которую должны решить пользователи имитационной среды, является начальное определение области предполагаемого моделирования, то определение того, что считать моделируемой организацией. Эта задача требует понимания бизнес-процессов, для участия в которых предполагается использовать разрабатываемую систему. В режиме адаптации, следовательно, необходимо определить и формализовать бизнес-процессы данной организации.

В режиме эксперимента производится имитационное моделирование процессов, определенных в режиме адаптации. По выбору пользователей, модели компонентов разрабатываемой системы, зарегистрированных в имитационной среде, могут сопровождаться графической анимацией или выполняться как реальные компоненты. Возможно переопределение значений параметров, заданных на этапе адаптации. Кроме того, пользовате-

ли могут оставлять комментарии по поводу их впечатления от характеристик качества системы. Для этого предлагается дать им возможность формального оценивания зарегистрированных компонент разрабатываемой системы.

В режиме анализа пользователи анализируют данные, полученные в режиме эксперимента. Для таких пользователей (как правило, это инженеры по сбору требований), имитационная среда предоставляет ряд отчетов с анализом результатов экспериментов. Каждый такой отчет включает номер версии модели, а также набор оценок значений всех параметров качества, которые использовались в данном эксперименте.

2. Обзор пакетов приложения. Для реализации возможности создания и проведения экспериментов с имитационными моделями необходима разработка рабочей среды, взаимодействующей с пользователями посредством веб-интерфейса. Приложение разрабатывается на основе объектно-ориентированного подхода с использованием языка Java, обеспечившего возможность развертывания приложения практически на любой программно-аппаратной платформе и применением шаблона проектирования MVC – «модель-представление-контроллер» [5]. В качестве эталонной архитектуры приложения выбрана трехуровневая архитектура с тонким клиентом и выделенным сервером приложений.

Классы приложения распределены по пакетам, в соответствии с выполняемой бизнес-логикой. Диаграмма пакетов показана на рисунке 1. Для хранения данных как режима адаптации, так и режимов эксперимента и анализа используется реляционная база данных, разработанная средствами СУБД MySQL. Для подключения и работы с базой данных используются классы пакета pose.database.

Пакет pose.util содержит классы утилитарного характера, применяемые в различных частях системы независимо от функциональности, возложенной на эти части. Пакет pose.forms содержит классы java beans, необходимые для получения и отправки данных на веб-страницы приложения. Пакет pose.controller содержит классы контроллеров, управляющих потоком выполнения системы. Пакет pose.users используется для обеспечения функциональности системного администратора. Пакет pose.sim является основным пакетом подсистемы моделирования, так как содержит классы, предназначенные для работы с имитационными моделями. Пакет pose.process является основным пакетом подсистемы формирования и сопровождения моделей, содержит классы, необходимые для работы с моделями бизнес-процессов и моделью организации.

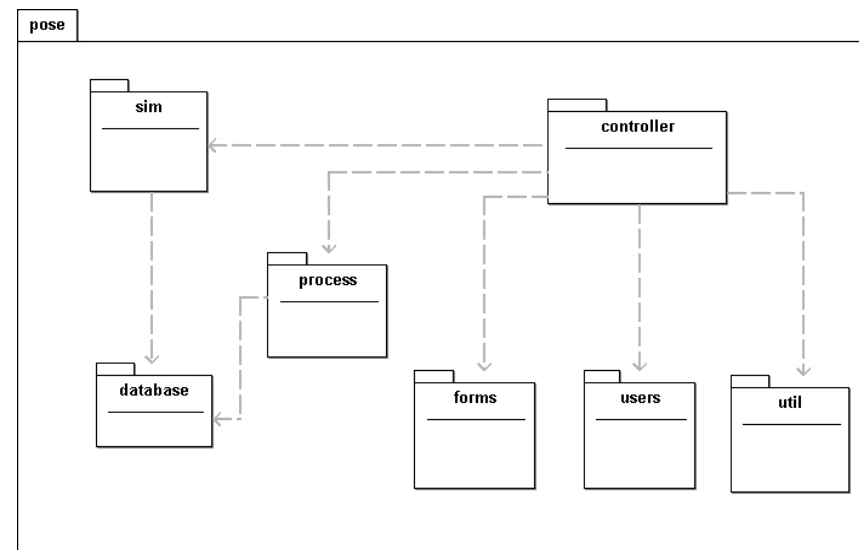


Рис. 1. Пакеты интерактивной имитационной среды

3. Реализация подсистемы формирования и поддержки моделей. Среди основных функций системы, относящихся к этапу адаптации, можно выделить формирование процессов использования и сборку имитационной модели [6]. Эти функции реализованы классами пакета pose.process.

Первым этапом работы является получение моделей процессов использования с ассемблера посредством интерфейса веб-сервисов. Полученные модели процессов использования регистрируются в системе и становятся доступными для дальнейшего использования. Для работы с моделями процессов в системе разработан класс Process.

К моделям процессов, размещенным в хранилище системы, могут быть подключены компоненты процессов использования, несущие в себе функциональность моделируемой системы. Они могут быть компонентами как разрабатываемой системы, так и уже существующей. Для работы с компонентами процессов использования в системе предусмотрен класс ProcessComponent. Модели процессов также связываются с определенными подразделениями организации, для чего в пакете pose.process предусмотрен класс OrganisationUnit. Модели процесса использования могут быть присвоены метки: «хорошо», «плохо», «нейтрально».

После завершения работ по формированию моделей процессов, пользователи системы определяют наборы процессов использования и связы-

вают их с моделью организации. Для выполнения этих действий используется классы ProcessSet и OrganisationStructure. Набору процессов использования могут быть присвоены метки «как есть» или «на будущее»

Завершающим этапом режима адаптации является формирование имитационной модели. На этом этапе пользователь системы определяет, какие наборы процессов использования, подразделения организации будут входить в модель, в каком формате будет представлена модель. Данный процесс является итеративным, пользователь пошагово может уточнять модель, разрешать возникающие конфликтные ситуации. После завершения определения набора объектов, составляющих модель, система формирует физическое представление модели, в соответствии с выбранным форматом и загружает модель в среду выполнения. Таким образом, итогом данного этапа является готовая к выполнению модель, развернутая в среде выполнения. В последствии модель может быть отредактирована, собрана и снова загружена в среду выполнения. Функциональность, отвечающая за формирование и сборку модели, реализована в классе ModelBuilder.

4. Реализация подсистемы моделирования. К основным функциям подсистемы моделирования относятся организация процесса оценивания качества и непосредственно оценивание качества [7]. Реализация этих функций осуществляется при помощи классов пакета pose.sim.

Первым этапом работы системы является параметризация модели. При этом выделяется набор оцениваемых характеристик качества, каждой из которых ставится в соответствие некий параметр, используемый во время моделирования, и задаются начальные значения этих параметров. Для проведения этих действий в системе предусмотрен класс Parameters.

Управление работой имитационной модели осуществляется при помощи класса Simulation, содержащего методы, необходимые для запуска и остановки модели, создания сессии моделирования и начала взаимодействия заинтересованных сторон с выполняемой моделью. В процессе работы модели с целью проведения экспериментов к ней могут подключаться различные пользователи, для каждого из которых создается начальный набор оценок характеристик качества, которые затем будут изменены пользователем при помощи методов класса Assessments. Работа заинтересованного лица с моделью сводится к выполнению управляющих воздействий в соответствии с логикой работы системы, а также периодическое оценивание характеристик качества путем обработки специальных механизмов, названных «перехватчиками».

Итогом работы пользователя с моделью является отчет о результатах эксперимента, содержащий набор пар «значение параметра – оценка пара-

метра», формируемый на основе объектов классов Parameters и Assessments, и обрабатываемый при помощи методов класса Report. Полученные оценки качества могут быть расценены как удовлетворительные и стать основой для требований качества, или неудовлетворительные, что приведет к необходимости проведения дополнительных экспериментов.

5. Выводы. При организации процесса разработки прототипа интерактивной имитационной среды было проведено разделение процесса сбора требований качества на отдельные этапы: адаптация моделей, эксперимент и анализ результатов. Было принято решение реализовывать каждый из этих этапов при помощи отдельного набора классов. Для режима адаптации рассмотрены принципы функционирования подсистемы формирования и сопровождения имитационной модели, начиная от получения процессов функционирования системы от ассемблера процессов и заканчивая формированием работоспособной имитационной модели. Для режимов эксперимента и анализа результатов рассмотрены принципы выполнения моделей, взаимодействия с ними заинтересованных сторон, получение на основании такого взаимодействия пользовательских оценок качества, а также последующий анализ пользовательских оценок с целью выявления и формализации требований качества.

При развитии прототипа планируется добавить поддержку новых форматов представления имитационных моделей, применить к разработанной системе технологию AJAX [8], позволяющую улучшить качество пользовательского представления данных и снизить нагрузку на сеть.

Список литературы: 1. Методы получения требований. <http://www.caseclub.ru/articles/trebmethod.html>, 20.06.2007. 2. Wieggers K. Writing Quality Requirements // <http://www.processimpact.com/articles/qualreqs.html>, 29.12.2006. 3. Шеховцов В.А., Кашек Р., Златкин С.А., Долгарев А. О сборе требований качества с использованием имитационного моделирования // Вестник Национального технического университета "ХПИ". Системный анализ, управление и информационные технологии-2007.—№7.—С.17-31. 4. Shekhovtsov, V.; Kaschek, R.; Zlatkin, S. Constructing POSE: a Tool for Eliciting Quality Requirements. // Proc. ISTA'2007, LNI P-107, GI-Edition, 2007. — P. 187-199. 5. Фаулер М. Архитектура корпоративных программных приложений.— М.: Вильямс, 2006. 6. Векишин А.В. Разработка архитектурных моделей и программных решений формирования имитационных моделей для системы сбора требований качества // Бак. работа. - Харьков.: НТУ «ХПИ», 2007. 7. Семенюк А.Д. Разработка архитектурных моделей и программных решений имитационного моделирования бизнес-процессов для системы сбора требований качества // Бак. работа. - Харьков.: НТУ «ХПИ», 2007. 8. Крейн Д., Паскарелло Э. Ajax в действии.— М.: Вильямс, 2006.

Поступила в редколлегию 26.02.08